Having thus described our invention, we now claim:

1. A method for administration and replication of a database, comprising the steps of:

providing a database management system with a built-in random sampling facility integrated into said database management system; and,

executing said random sampling facility from within the database management system to perform a replication operation on said database.

2. The method as set forth in claim 1, further comprising the steps of:

defining a database record sample size S;

randomly sampling S records of the database using said random sampling facility;

storing statistics for each of said S records, wherein said statistics include a record key for each record; and,

producing an extrapolated replication partition analysis based on said statistics.

3. The method as set forth in claim 2, wherein the step of defining said sample size S includes:

defining a default sample size;

selectively receiving a desired sample size; and, setting said sample size S as said default sample size when the desired sample size is not selectively received, and setting said sample size S as said desired sample size when the desired sample size is selectively received.

4. The method as set forth in claim 1, further comprising the steps of:

defining a database record sample size S;

randomly sampling S records of the database using said random sampling facility;

storing statistics for each of said S records, wherein said statistics include a record key for each record; and,

producing a partial replication partition analysis based on said statistics.

5. The method as set forth in claim 4, wherein the step of defining said sample size S includes:

defining a default sample size;

selectively receiving a desired sample size; and,

setting said sample size S as said default sample size when the desired sample size is not selectively received, and setting said sample size S as said desired sample size when the desired sample size is selectively received.

6. A method for database administration and replication, comprising the steps of:

providing a database management system with an integrated random sampling facility;

selecting a default sample size value S;

selectively receiving a desired sample size value D and setting said default sample size value S to said desired sample size value D when said desired sample size value D is received;

randomly sampling S records of the database using said random sampling facility;

storing statistics for each of said S records, wherein said statistics include a record key for each record; and,

producing at least one of:

an extrapolated replication partition analysis based on said statistics; and

a partial replication partition analysis based on said statistics.

7. The method as set forth in claim 6, wherein the step of selecting said default sample size value D further includes the steps of:

generating a table of S number pairs (Y_j,I_j), j=1,2,...,S, wherein all Y and all I are initially set to zero;

initializing a reservoir of records to an empty +state;

setting an index M to said reservoir equal to zero;

generating a sequence of N non-repeating random numbers $U_1,U_2,...,U_N$,

$0 \leq U \leq 1$, wherein N is the number of records in the database; and,

performing additional steps for each random number U_k generated, k=1,2,...,N, the additional steps including:

skipping the next record in the database if U_k is less than the smallest value of Y in said table of number pairs; and,

updating the table if a Y less than U_k exists by performing further steps including:

setting M equal to its current value plus one;

replacing the smallest Y in the table with U_k;

setting the I value paired with the smallest Y equal to M; and,

storing all or part of the next record of the database in said reservoir of stored records, wherein the current value of M is a reservoir index to said stored record.

8. The method as set forth in claim 7, wherein the step of updating the table further includes the step of:

arranging the table in a heap with respect to Y.

9. The method as set forth in claim 6, further comprising the step of:

sorting said stored statistics by key prior to producing said partition analysis.

10. The method as set forth in claim 9, further comprising the steps of:

accessing all database records in an arbitrary sequence;

iteratively filling all of said partitions except the last said partition with said accessed records to a maximum byte count; and,

storing remaining accessed records in the last of said partitions.

11. The method as set forth in claim 6, wherein the step of storing statistics includes storing said statistics in a memory.

12. The method as set forth in claim 11, wherein the step of storing statistics includes storing said statistics in said memory in a compressed format.

13. The method as set forth in claim 6, wherein the step of producing at least one of said partition analyses includes the step of defining multiple partition boundaries.

14. The method as set forth in claim 6, wherein the step of sampling said S records includes randomly sampling the S records utilizing dataspaces including:

at least one index dataspace;

at least one key dataspace; and,

at least one statistics dataspace.

15. A database management system (DBMS) for managing an associated database, the DBMS comprising:

random sampling facility integrated with the database management system;

first database analysis tools using said integrated random sampling facility for generating extrapolated reports on database content;

second database analysis tools using said integrated random sampling facility for generating extrapolated reports on database size; and,

database replication tools adapted to execute at least one of a complete replication having output partition sizes determined by extrapolating a random sample of said database, and a partial replication in which the data stored in the partial replication comprises a random sample of said database.

16. The database management system of claim 15 further comprising:

a pre-configured number S defining a default sample size;

a means for selectively receiving a particular number defining a desired sample size and setting said number S equal to said particular number;

a means for randomly sampling S records of the database using said random sampling facility;

a means for storing statistics for each of said S records, wherein said statistics include a record key for each record; and,

a means for producing at least one of:

an extrapolated database content analysis based on said statistics;

an extrapolated partition analysis based on said statistics; and,

a partial partition analysis based on said statistics.

17. The database management system of claim 16, further comprising:

a means for sorting said stored statistics by key prior to producing at least one of said analyses.

18. The database management system of claim 16, wherein said means for randomly sampling S records further comprises:

a means for generating a table of S number pairs (Y_j,I_j), $j=1,2,\ldots,S$, wherein all Y and all I are initially zero;

a means for initializing a reservoir of records to an empty state;

a means for setting an index M to said reservoir equal to zero;

a means for generating a sequence of N non-repeating random numbers $U_1,U_2,\ldots,U_N$, $0\leq U\leq 1$, wherein N is the number of records in the database; and,

a means, for each random number U_k generated, $k=1,2,\ldots,N$, comprising:

a means to skip the next record in said database if U_k is less than the smallest value of Y in said table of number pairs; and,

a means to update the table if a Y less than U_k exists, comprising:

a means to set M equal to its current value plus one;

a means to replace the smallest Y in the table with U_k;

a means to set the I value paired with the smallest Y equal to M; and,

a means to store all or part of the next record of said database in said reservoir of stored records, wherein the current value of M is a reservoir index to said stored record.

19. The database management system of claim 18 wherein the means to update the table further comprises:

a means to arrange the table in a heap with respect to Y.

20. The database management system of claim 18, wherein said means for storing statistics comprises a means for storing said statistics in memory.

21. The database management system of claim 20, further comprising a means for sorting said stored statistics by key prior to producing at least one of said analyses.

22. The database management system of claim 21, wherein said partition analyses include analyses of multiple partition boundaries.

23. The database management system of claim 22, further comprising:

a means for accessing all database records in an arbitrary sequence;

a means for iteratively filling all of said partitions except the last with said accessed records to a maximum byte count; and,

a means for storing remaining accessed records in the last of said partitions.

24. The database management system of claim 16, further comprising:

a means for utilizing at least one index dataspace;

a means for utilizing at least one key dataspace; and,

a means for utilizing at least one statistics dataspace.